

Feudal

Relatório Intercalar



Universidade do Porto

Faculdade de Engenharia

FEUP

Mestrado Integrado em Engenharia Informática e
Computação

Programação em Lógica

Grupo 01:

Jorge Miguel Marques Reis - 080509053

Miao Sun - 080509162

Faculdade de Engenharia da Universidade do Porto
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

6 de Outubro de 2012

Resumo

Neste relatório pretende-se apresentar com detalhe o jogo de tabuleiro “Feudal”, e abordar a implementação de uma aplicação para o jogo em linguagem de Programação Prolog.

1 Introdução

Este trabalho tem como objectivo adquirir conhecimentos de implementação segundo o Paradigma da Programação Lógica, desenvolvendo em linguagem de programação Prolog, uma aplicação que permita jogar um jogo tradicional de tabuleiro. Como tal, pretende-se utilizar conceitos lógicos para a recriação de jogadas e implementação da mecânica de jogo segundo as regras de movimentação das peças, e a determinação do fim do jogo com atribuição de um vencedor. O Feudal tem semelhanças ao Xadrez, mas utilizando um mapa com certos terrenos que restringem o movimento. É um jogo para duas pessoas, preparado também para equipas, até 6 jogadores, com regras ligeiramente distintas. No contexto do objectivo deste trabalho, apenas se vai implementar o modo de dois jogadores, mas que permite três tipos de utilização: Humano/humano, Humano/Computador e Computador/Computador. Assim, é necessário desenvolver métodos para representar o estado do jogo em cada jogada, visualizando o tabuleiro, e permitir a interacção dos jogadores com o mesmo. Será também necessário incluir métodos de inteligência artificial, que permitam desenvolver diversos níveis de jogo para o computador.

O trabalho aqui apresentado está a ser desenvolvido utilizando os interpretadores de Prolog: Swi-Prolog e Sictus Prolog.

As proximas secções deste relatório apresentam:

- A descrição do problema, história e regras do jogo;
- Representação do estado do jogo em Prolog;
- Representação de movimentos de jogo em Prolog;
- Visualização do Tabuleiro em modo de texto com predicados em Prolog;
- Conclusões e Perspectivas para o desenvolvimento do trabalho;
- Código desenvolvido até ao momento.

2 Descrição do Problema

O trabalho baseia-se na implementação do jogo de tabuleiro “Feudal” para computador, utilizando a linguagem de programação Prolog. O feudal é um jogo estilo Xadrez, originalmente lançado pela 3M Company em 1967. Utilizando recriações de personagens medievais, o objectivo do jogo é invadir e tomar o castelo do adversário ou matar a sua realeza, O Rei, o Príncipe e o Duque, enquanto estrategicamente protege o seu próprio castelo e realeza. O tabuleiro de jogo completo é composto por uma área com 24 por 24 posições (ver Figura 1), dividido a meio por um separador no início do jogo para que cada jogador

posicione as suas peças secretamente na sua metade do tabuleiro, respeitando certas regras que serão descritas mais à frente.

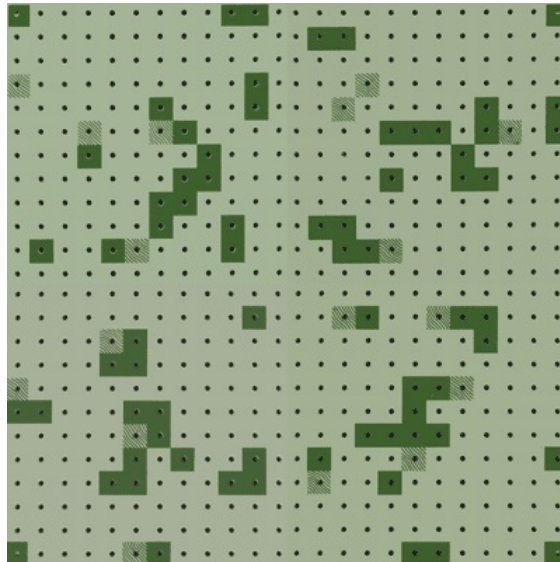


Figura 1: Tabuleiro de Jogo

Cada jogador possui um conjunto de 14 figuras:

- 1 Castle
- 1 King
- 1 Prince
- 1 Duke
- 2 Knights
- 2 Sergeants
- 1 Squire
- 1 Archer
- 4 Pikemen

Ao colocar as peças no tabuleiro, o jogador tem de ter atenção a alguns pormenores. As montanhas e terreno acidentado são áreas restritas de jogo. Antes de posicionar os peças, o jogador deve verificar as regras de posicionamento e movimentação para homens montados em cavalos e homens apeados. O castelo pode ser colocado em qualquer posição do seu território, incluindo montanhas e terrenos acidentados. Uma vez posicionado, o castelo não pode ser movido. É necessário também definir uma posição à volta do castelo que consistirá no único ponto de entrada ou saída do mesmo. Homens a cavalo não podem ser colocados nem mover para ou através de montanhas ou terrenos acidentados porém, podem atravessar vales entre os mesmos. Homens apeados também não podem ser colocados nessas zonas especiais, contudo podem atravessar terrenos

acidentados e vales entre montanhas. Por último, Archers e Squires são as únicas peças que não podem estar no seu castelo, no entanto podem ser colocados ou movidos sobre a posição de entrada do mesmo.

Pode ser usada a técnica de moeda ao ar para definir que começa primeiro. Durante a sua jogada, cada jogador pode mover quaisquer ou até todas as suas peças, mas pelo menos uma peça tem que ser movida. Cada peça apenas pode ser movida uma vez por turno, e deverá ser movida consoante as regras de cada peça (*ver Figura 2 - Diagrama de Movimentos*). Apenas Squires podem ser movidos sobre espaços ocupados, excepto pelo castelo, e todos os outros homens apenas podem atravessar espaços desocupados. Quando se move para atacar/matar, a peça atacante (excepto Archer) deve mover-se para o espaço ocupado pela peça do inimigo, que sairá do tabuleiro de jogo. Os Archers, em vez disso, podem disparar sobre o primeiro homem numa linha de fogo até três posições de distância em qualquer direcção. No entanto, não podem disparar sobre montanhas ou castelos. Para atravessar a posição de entrada do Castelo, ou para sair ou entrar do castelo, um homem deve parar nessa posição e esperar pelo próximo turno para deixá-la. Note-se que o movimento do Squire torna impossível a este entrar no Castelo inimigo a partir da posição de entrada. O jogo termina quando o Castelo do adversário é tomado, ou quando toda a sua realeza é assassinada. Para tomar o Castelo, o invasor deve chegar à posição de entrada do Castelo e no próximo passo entrar e ganhar o jogo.

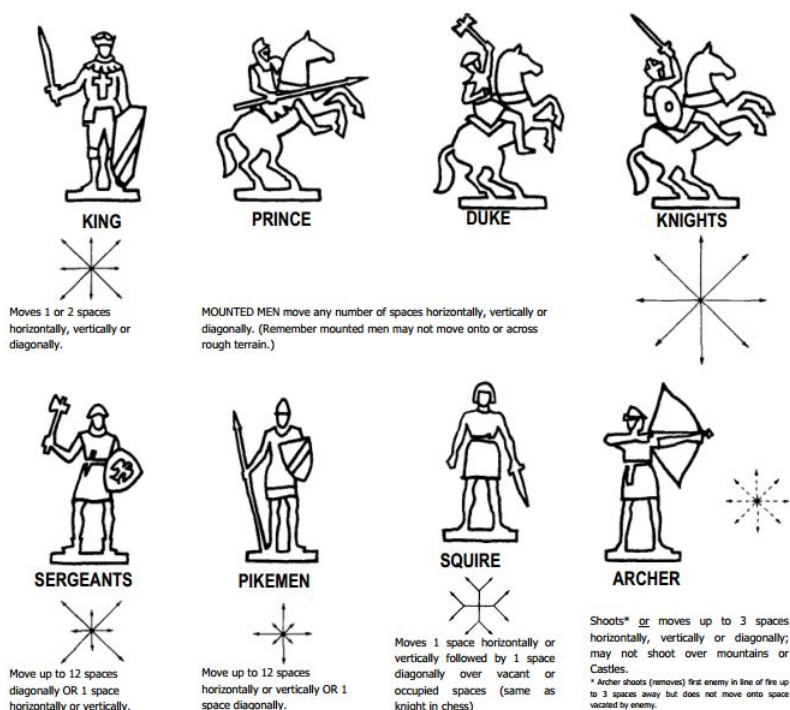


Figura 2: Diagrama de Movimentos

3 Representação do Estado do Jogo

Este jogo desenrola-se num tabuleiro quadrado de 24 por 24 posições, como demonstrado na figura 1, e como tal optou-se por definir o mesmo em Prolog como uma lista de listas.

4 Representação de um Movimento

5 Visualização do Tabuleiro

6 Conclusões e Perspectivas de Desenvolvimento

Bibliografia

- [1] Autor(es). Nome da referencia. <http://www.fe.up.pt>, Ano da pagina. Online em Setembro 2011.
- [2] Donald E. Knuth. *The T_EXbook*. Addison-Wesley, 1984.

A Código Desenvolvido